

Тема 7. Часть 2

ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ

Вопросы:

3. СРЕДСТВА АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ ИС. CASE-СИСТЕМЫ
4. ОЦЕНКА КАЧЕСТВА ИНФОРМАЦИОННОЙ СИСТЕМЫ. КРИТЕРИИ КАЧЕСТВА ИС
5. РЕИНЖИНИРИНГ И ЕГО МЕСТО В ЖЦ ИС. МЕТОДЫ И ТЕХНОЛОГИИ РЕИНЖИНИРИНГА ИС

Средства автоматизации проектирования ИС. CASE- системы

Технология проектирования ИС –
совокупность методов и средств
проектирования ИС, а также
организации и управления,
внедрения и модернизации проекта
информационной системы

Средства проектирования ИС

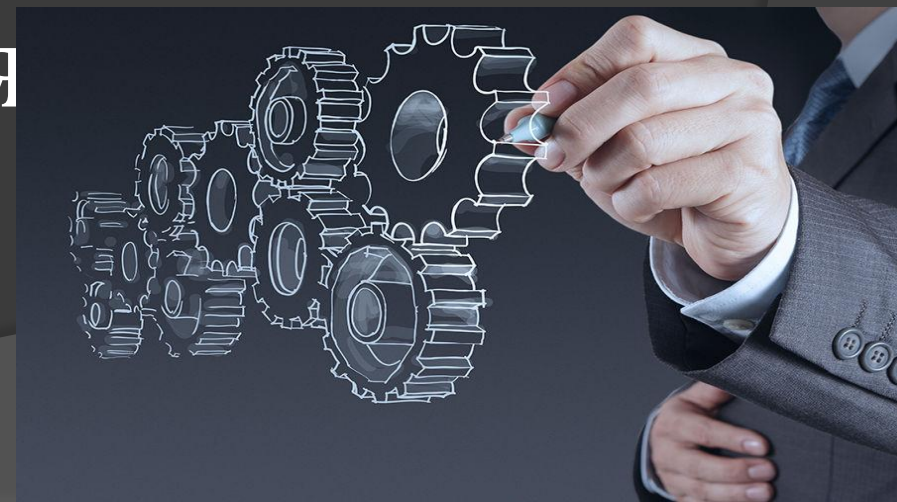
- ◎ нормативно-правовые документы
(стандарты, руководящие документы)
- ◎ системы классификации и кодирования информации
- ◎ системы проектной документации

Средства проектирования ИС

- модели ИС и их компонентов
- методики анализа и принятия проектных решений
- программные средства

Средства автоматизации проектирования ИС. CASE-системы — инструменты автоматизации процессов проектирования и разработки программного обеспечения для системного аналитика, разработчика ПО

и программиста



CASE-средства

CASE (англ. *Computer-Aided Software Engineering*) – программные средства, поддерживающие процессы создания и сопровождения ИС, включая анализ и формулировку требований, проектирование прикладного ПО и баз данных, тестирование, документирование, обеспечение качества, управление проектом, а также другие процессы.

CASE (англ. *Computer-Aided Software Engineering*) — набор инструментов и методов программной инженерии для проектирования программного обеспечения, который помогает обеспечить высокое качество программ, отсутствие ошибок и простоту в обслуживании программных продуктов

Виды современных CASE-средств

- ◎ простые средства анализа и документирования
- ◎ полномасштабные средства автоматизации, покрывающие весь жизненный цикл ПО

Характерные особенности CASE-средств:

- мощные графические средства для описания и документирования ИС, обеспечивающие удобный интерфейс с разработчиком и развивающие его творческие возможности;

Характерные особенности CASE-средств:

- интеграция отдельных компонент CASE-средств, обеспечивающая управляемость процессом разработки ИС;

Характерные особенности CASE-средств:

- использование специальным образом организованного хранилища проектных метаданных (репозитория);

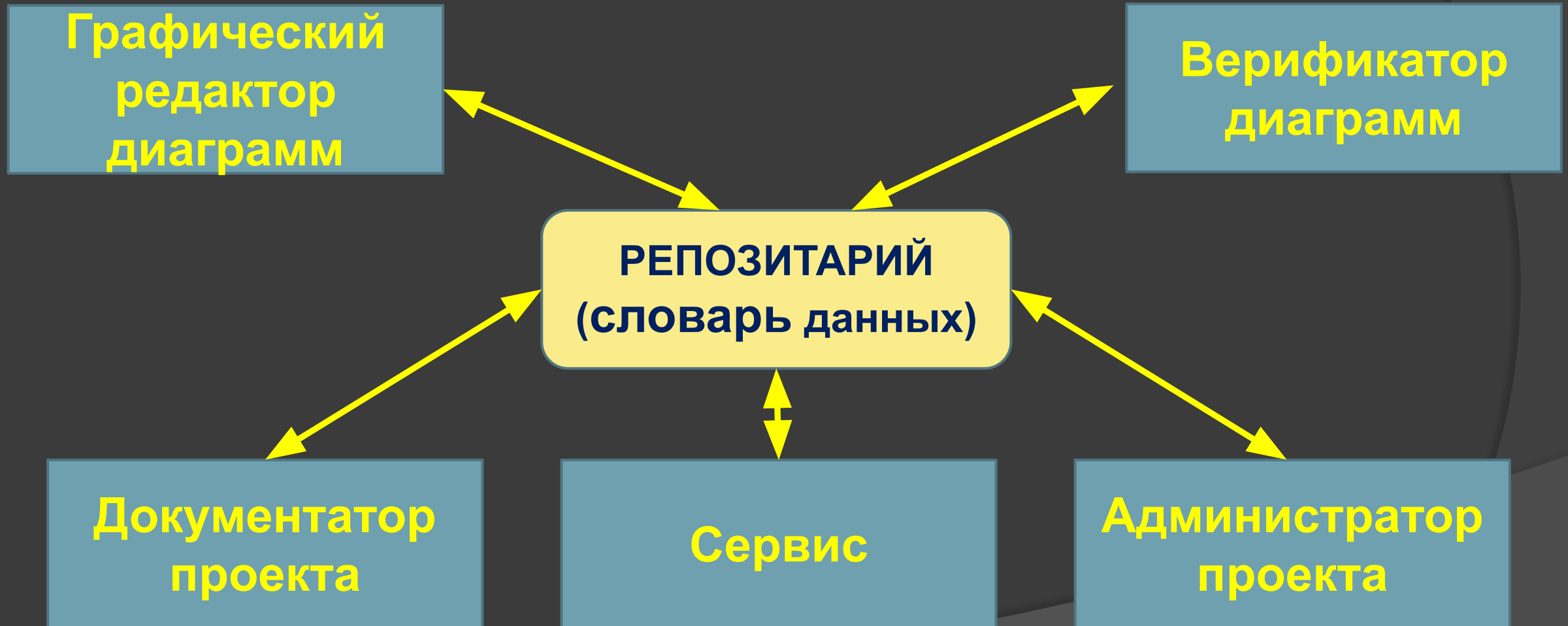
Компоненты интегрированной CASE-системы (комплекс средств, поддерживающих полный ЖЦ ПО)

- ◎ репозитарий
- ◎ графические средства анализа и проектирования;
- ◎ средства разработки приложений,

Компоненты интегрированной CASE-системы

- средства конфигурационного управления;
- средства документирования;
- средства тестирования;
- средства управления проектом;
- средства реинжиниринга

Архитектура CASE-системы



Архитектура CASE-системы

Ядром системы является — репозиторий.

Он представляет собой специализированную базу данных, предназначенную для отображения состояния проектируемой ИС в каждый момент времени.

Репозиторий содержит информацию об объектах проектируемой ИС и взаимосвязях между ними, все подсистемы обмениваются данными с ним.

В репозитории хранятся описания следующих объектов:

- проектировщиков и их прав доступа к различным компонентам системы;
- организационных структур;
- диаграмм; компонентов диаграмм; связей между диаграммами;
- структур данных; программных модулей; процедур; библиотеки модулей и т.д.

Графические средства моделирования предметной области позволяют разработчикам автоматизированных ИС в наглядном виде

- изучать существующую информационную систему,
- перестраивать ее в соответствии с поставленными целями и имеющимися ограничениями.

Все модификации диаграмм, выполняемых разработчиками в интерактивном (диалоговом) режиме, вводятся в словарь данных, контролируются с общесистемной точки зрения и могут использоваться для дальнейшей генерации действующих функциональных приложений. В любой момент времени диаграммы могут быть распечатаны для включения в техническую документацию проекта.



Архитектура CASE-системы

Графический редактор выполняет операции:

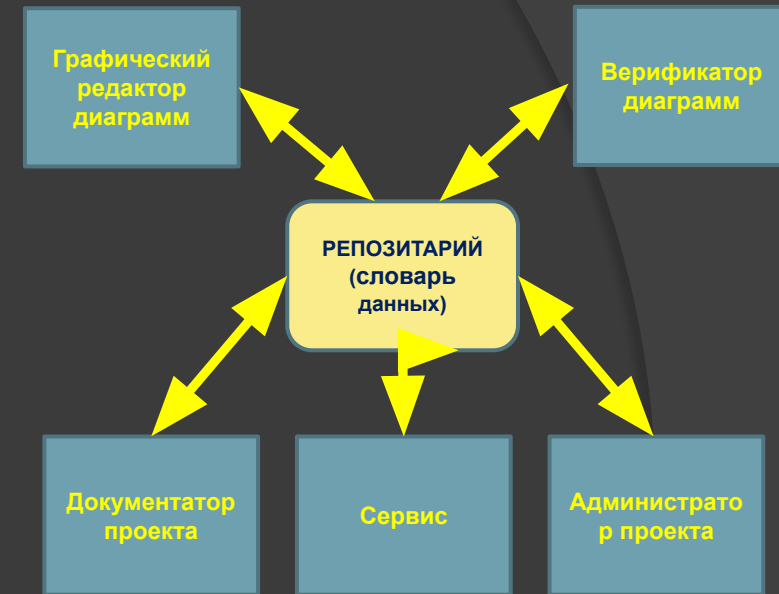
- ⦿ создание элементов диаграмм и взаимосвязей между ними;
- ⦿ задание описания элементов диаграмм;
- ⦿ задание описания связей между элементами диаграмм;
- ⦿ редактирование элементов диаграмм, их взаимосвязей и описаний.



Архитектура CASE-системы

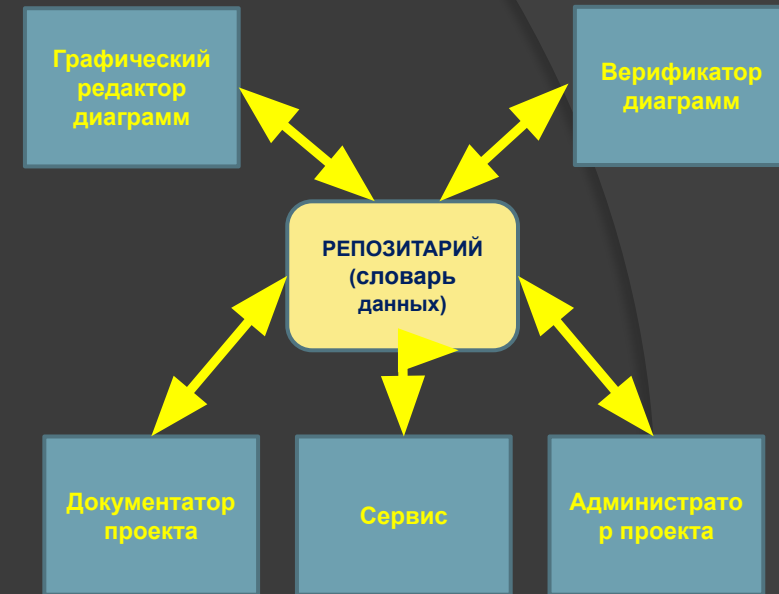
Верификатор диаграмм служит для контроля правильности построения диаграмм в заданной методологии проектирования ИС, выполняет функции:

- ⦿ мониторинг правильности построения диаграмм;
- ⦿ диагностику и выдачу сообщений об ошибках;
- ⦿ выделение на диаграмме ошибочных элементов.



Архитектура CASE-системы

Документатор проекта позволяет получать информацию о состоянии проекта в виде различных отчетов, которые могут строиться по нескольким признакам (времени, автору, элементам диаграмм, диаграмме или проекту в целом)



Архитектура CASE-системы

Администратор проекта представляет собой инструменты, необходимые для выполнения следующих административных функций:

- ⦿ инициализации проекта;
- ⦿ задания начальных параметров проекта;
- ⦿ назначения и изменения прав доступа к элементам проекта;
- ⦿ мониторинга выполнения проекта.



Архитектура CASE-системы

Сервис – набор системных утилит по обслуживанию репозитария, которые выполняют функции архивации данных, восстановления данных и создания нового репозитария.



Для CASE существенны

4 типа диаграмм:

- ⦿ функционального проектирования (DFD – диаграммы потоков данных),
- ⦿ моделирования данных (ERD – диаграммы «сущность-связь»),
- ⦿ моделирования поведения (STD – диаграммы переходов состояний),
- ⦿ структурные (карты), применяющиеся на этапе проектирования и описывающие отношения между модулями и внутримодульную структуру.

Современные диаграммеры обеспечивают:

- ◎ создание иерархически связанных диаграмм, в которых комбинируются графические и текстовые объекты;
- ◎ создание и редактирование объектов в любом месте диаграммы;
- ◎ создание, перемещение и выравнивание групп объектов, изменение их размеров, масштабирование;
- ◎ сохранение связей между объектами при их перемещении и изменении размеров;
- ◎ автоматический контроль ошибок и др.

Основные классификационные признаки CASE-средств:

- по категориям;
- по типам

Классификация по категориям определяет степень интегрированности по выполняемым функциям:

- ◎ локальные средства, решающие небольшие автономные задачи (tools),
- ◎ набор частично интегрированных средств, охватывающих большинство этапов жизненного цикла ИС (toolkit)
- ◎ полностью интегрированные средства, поддерживающие весь ЖЦ ИС и связанные общим репозиторием

Классификация по типам отражает функциональную ориентацию CASE-средств на те или иные процессы ЖЦ, в основном совпадает с компонентным составом CASE-средств и включает следующие основные типы:

- ◎ средства анализа (Upper CASE),
- ◎ средства анализа и проектирования (Middle CASE)
- ◎ средства проектирования баз данных
- ◎ средства разработки приложений
- ◎ средства реинжиниринга

◎ **Средства анализа (Upper CASE)**
предназначены для построения и
анализа моделей предметной области
(**Design/IDEF, VPwin**);

◎ Средства анализа и проектирования (Middle CASE)

поддерживают наиболее распространенные методологии проектирования и используются для создания проектных спецификаций (**Vantage Team Builder, Designer/2000 – ORACLE, Business Studio – Современные технологии управления**).

Выходом таких средств являются спецификации компонентов и интерфейсов системы, архитектуры системы, алгоритмов и структур данных.

◎ Средства проектирования БД
входят в состав (DataBase Designer –
ORACLE, AllFusion ERwin Data Modeler
(ERWin) – Computer Associates
Technologies).

◎ Средства разработки приложений

К ним относятся средства 4GL (**Uniface, JAM, PowerBuilder, Developer/2000, New Era** и др.) и генераторы кодов, входящие в состав **Vantage Team Builder** и др.

◎ Средства реинжиниринга

обеспечивают анализ программных кодов и схем баз данных и формирование на их основе различных моделей и проектных спецификаций.

Помимо этого, **CASE-средства можно классифицировать по следующим дополнительным признакам:**

- ⦿ применяемым методологиям и моделям систем и БД;
- ⦿ степени интегрированности с СУБД;
- ⦿ доступным платформам



Вспомогательные типы включают:

- средства планирования и управления проектом (SE Companion, Microsoft Project и др.);
- средства конфигурационного управления (PVCs (Intersolv));
- средства тестирования (Quality Works (Segue Software));
- средства документирования (SoDA (Rational Software)).

Преимущества CASE-технологии по сравнению с традиционной технологией проектирования:

- Улучшение качества разрабатываемого программного приложения за счет средств автоматического контроля и генерации;
- Возможность повторного использования компонентов разработки;
- Поддержка адаптивности и сопровождения ИС;
- Снижение времени создания системы, что позволяет на ранних стадиях проектирования получить прототип будущей системы и оценить его;
- Освобождение разработчиков от рутинной работы по документированию проекта, так как используется встроенный документатор;
- Возможность коллективной разработки ИС в режиме реального времени.

Стратегия выбора CASE-систем для конкретного применения зависит как от целей и потребностей самого проекта, так и от квалификации вовлеченных в процесс проектирования специалистов.

При выборе CASE-системы учитываются следующие аспекты:

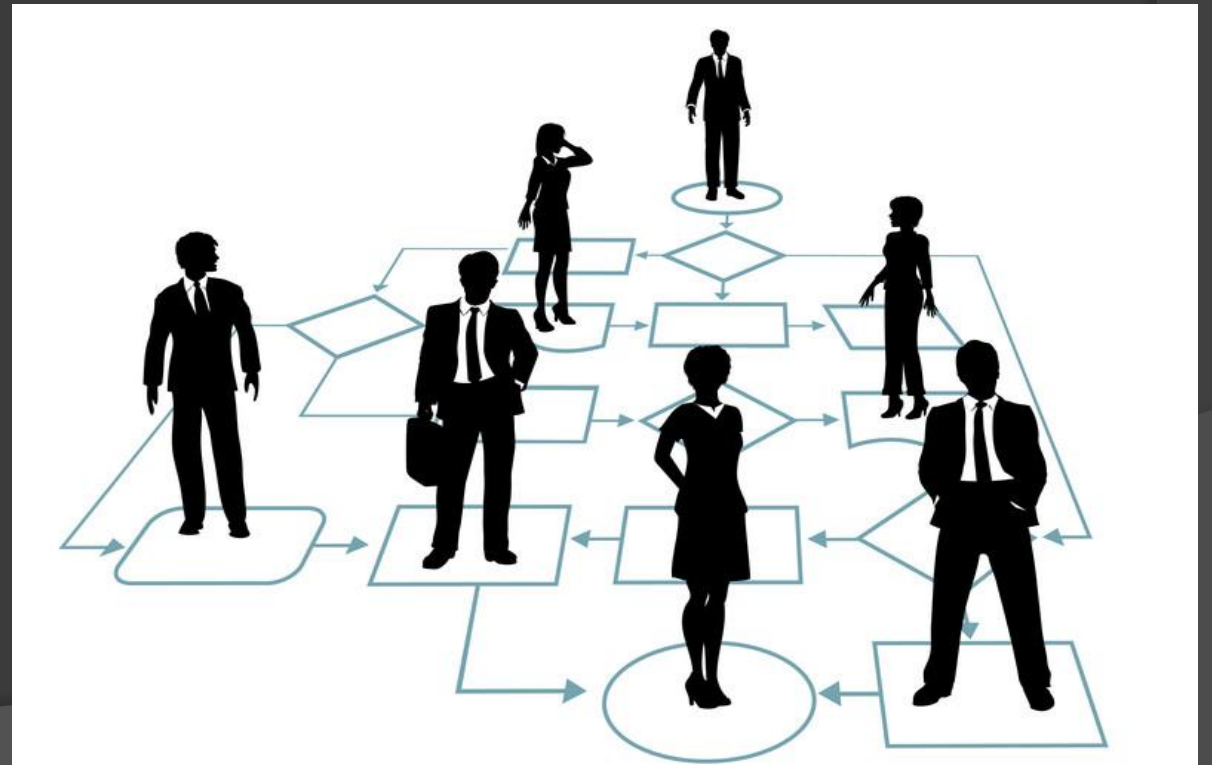
- Наличие базы проектных данных, архива или словаря;
- Интерфейсы с другими CASE-системами;
- Возможности экспорта/импорта;
- Многопользовательский режим;
- Открытая архитектура;
- Расширение новыми методологиями;
- Наличие графических средств поддержки методологий проектирования;

При выборе CASE-системы учитываются следующие аспекты:

- ⦿ Обеспечение качества проектной документации;
- ⦿ Автоматическая генерация отчетов о проектных решениях;
- ⦿ Решения (спецификации), созданные в процессе проектирования, служат источником документирования системы;
- ⦿ Генерация кодов программ;
- ⦿ Планирование и управление проектом.

На сегодняшний день рынок программного обеспечения СНГ располагает следующими наиболее развитыми CASE-средствами:

- ◎ Vantage Team Builder (Westmount I-CASE);
- ◎ Designer/2000;
- ◎ Silverrun;
- ◎ ERwin+VPwin;
- ◎ S-Designer;
- ◎ CASE.АНАЛИТИК;
- ◎ Business Studio



**Оценка качества
информационной системы.
Критерии качества ИС**

Оценка качества информационной системы. Критерии качества ИС

Оценка качества ИС - задача крайне сложная из-за многообразия интересов пользователей. Задачи оценки качества ИС являются близкими к задачам оценки качества программного обеспечения.

Качество программного обеспечения – способность программного продукта подтвердить свою спецификацию при условии, что спецификация ориентирована на характеристики, которые желает получить пользователь.

Качество ИС связано с дефектами, заложенными на этапе проектирования и проявляющимися в процессе эксплуатации ИС.

Свойства ИС, в том числе и дефектологические, могут проявляться лишь во взаимодействии с внешней средой, включающей технические средства, персонал, информационное и программное окружение.

В зависимости от целей исследования и этапов жизненного цикла ИС дефектологические свойства разделяют на :

- ⦿ дефектогенность,
- ⦿ дефектабельность
- ⦿ дефектоскопичность.



Дефектогенность определяется влиянием следующих факторов:

- численность разработчиков ИС, их профессиональные психофизиологические характеристики;
- условия и организация процесса разработки ИС;
- характеристики инструментальных средств и комплексов ИС;
- сложность задач, решаемых ИС;
- степень агрессивности внешней среды.

Дефектабельность характеризует наличие дефектов ИС и определяется их количеством и местонахождением.

Другими факторами, влияющими на дефектабельность, являются:

- структурно-конструктивные особенности ИС;
- интенсивность и характеристики ошибок, приводящих к дефектам.

Дефектоскопичность характеризует возможность проявления дефектов в виде отказов и сбоев в процессе отладки, испытаний или эксплуатации.

На дефектоскопичность влияют:

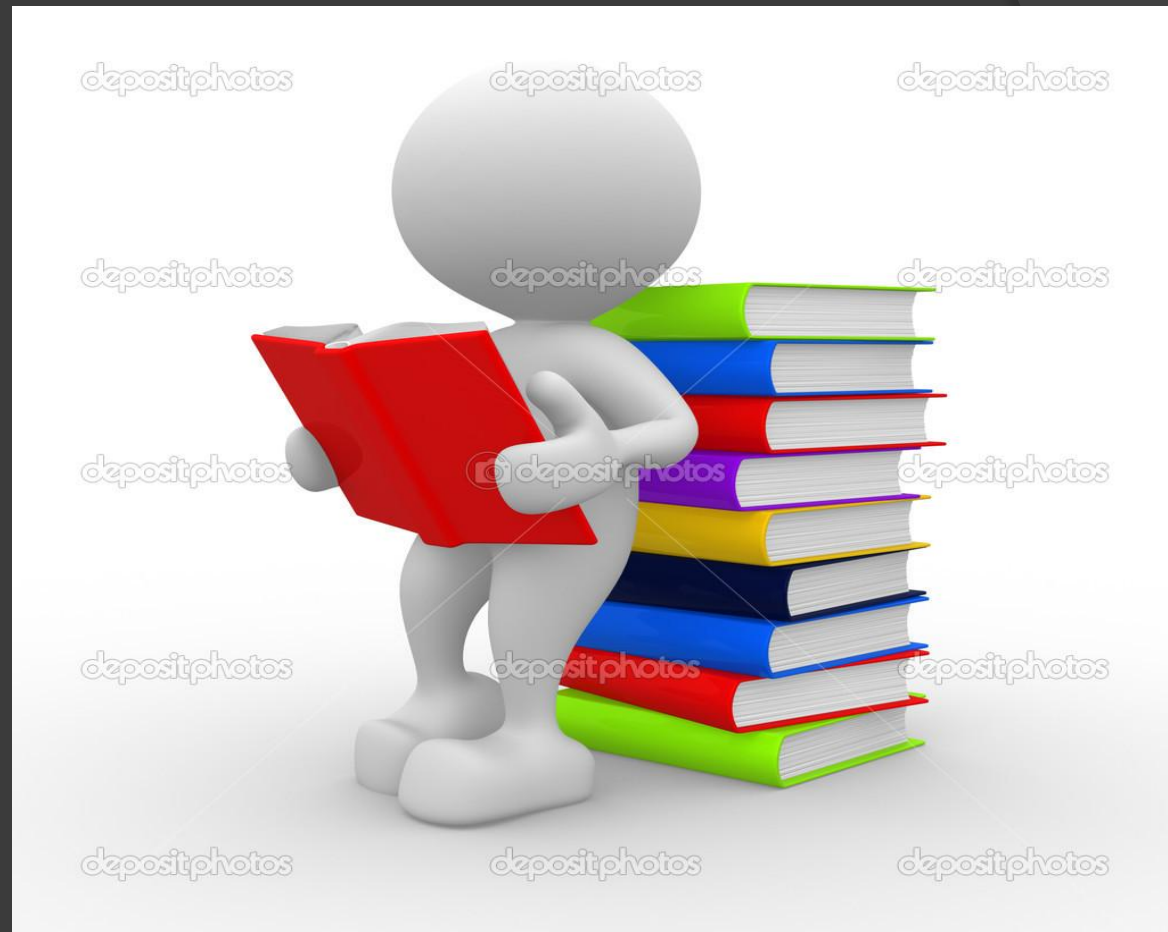
- количество, типы и характер распределения дефектов;
- устойчивость ИС к проявлению дефектов;
- характеристики средств контроля и диагностики дефектов;
- квалификация обслуживающего персонала.

Оценка качества ИС - задача крайне сложная из-за многообразия интересов пользователей.

Поэтому невозможно предложить одну универсальную меру качества и приходится использовать ряд характеристик, охватывающих весь спектр предъявляемых требований.

Показатели качества ИС:

- практичность;
- целостность;
- корректность;
- удобство обслуживания;
- оцениваемость;
- гибкость;
- адаптируемость;
- мобильность;
- возможность взаимодействия.



Каждому показателю качества ставится в соответствие группа критериев.

Один и тот же критерий может характеризовать несколько показателей

Сопоставление показателей и критериев качества ИС

- **практичность** - работоспособность, возможность обучения, коммуникативность, объем ввода, скорость ввода-вывода;
- **целостность** - регулирование доступа, контроль доступа;
- **эффективность** - эффективность использования памяти, эффективность функционирования;
- **корректность** - трассируемость, завершенность, согласованность;

Сопоставление показателей и критериев качества ИС

- ◎ **надежность** - точность, устойчивость к ошибкам, согласованность, простоту;
- ◎ **удобство обслуживания** - согласованность, простоту, краткость, информативность, модульность;
- ◎ **оцениваемость** - простоту, наличие измерительных средств, информативность, модульность;
- ◎ **гибкость** - распространяемость, общность, информативность, модульность;

Сопоставление показателей и критериев качества ИС

- ◎ **удобство обслуживания** - согласованность, простоту, краткость, информативность, модульность;
- ◎ **оцениваемость** - простота, наличие измерительных средств, информативность, модульность;
- ◎ **гибкость** - распространяемость, общность, информативность, модульность;

С помощью метрик можно дать количественную или качественную оценку качества ИС.

Различают следующие виды метрических шкал для измерения критериев.



- ◎ **Первый тип - метрики, которые используют интервальную шкалу,** характеризуемую относительными величинами реально измеряемых физических показателей, например, временем наработки на отказ, вероятностью ошибки, объемом информации и других.

◎ Второй тип - *метрики, которым соответствует порядковая шкала*, позволяющая ранжировать характеристики путем сравнения с опорными значениями.

◎ Третий тип - *метрики, которым соответствуют номинальная, или категорированная шкала*, определяющая наличие рассматриваемого свойства или признака у рассматриваемого объекта без учета градаций по этому признаку.

Так, например, *интерфейс* может быть "простым для понимания", "умеренно простым", "сложным для понимания".

Критерии качества информационных систем

Функциональные критерии оценивают степень выполнения ИС основных целей или задач.

Конструктивные критерии предназначены для оценки компонент ИС, не зависящих от целевого назначения.

Критерии качества информационных систем

Функциональные критерии качества информационных систем

Критерии этапа проектирования информационных систем

— Сложность

— Корректность

— Трудоемкость

Конструктивные критерии качества информационных систем

Критерии этапа эксплуатации информационных систем

— Сложность

— Надежность

— Эффективность

Факторы и параметры, влияющие на основные критерии качества

Критерии этапа сопровождения информационных систем

— Модифицируемость

— Мобильность

— Трудоемкость

Одним из факторов обеспечения качества ИС является

стандартизация – это установление и применение правил с целью упорядочения деятельности в определенных областях на пользу и при участии всех заинтересованных сторон, в частности, для достижения всеобщей оптимальной экономики при соблюдении функциональных условий и требований техники безопасности.

сертификация - деятельность по подтверждению соответствия продукта требованиям заказчика, а также государственным нормативным документам

Стандарты качества

Стандарт – технический нормативный правовой акт, разработанный в процессе стандартизации на основе согласия большинства заинтересованных субъектов технического нормирования и стандартизации и содержащий технические требования к продукции, процессам ее разработки, производства, эксплуатации (использования), хранения, перевозки, реализации и утилизации или оказанию услуг.

Стандарты нужны:

- ◎ **потребителям ИС** — для выбора техники, упорядочения деятельности и взаимодействия с поставщиками;
- ◎ **поставщикам продуктов и услуг** - для снижения себестоимости продукции и следования требованиям рынка;
- ◎ **разработчикам ИС** — для повышения качества решений и обеспечения совместимости с другими системами, применения повторно используемых решений, снижения трудоемкости и себестоимости работ, повышения их качества.

Рекомендации по оценке качества ИС

- ⦿ Для каждой характеристики качества рекомендуется формировать меры и шкалу измерений с выделением требуемых, допустимых и неудовлетворительных значений.
- ⦿ Реализация процессов оценки должна коррелировать с этапами жизненного цикла конкретного проекта программного средства в соответствии с применяемой, адаптированной версией стандарта ISO 12207.

Рекомендации по оценке качества ИС

- ◎ *Функциональная пригодность* – наиболее неопределенная и объективно трудно оцениваемая характеристика программного средства.

Области применения, номенклатура и функции комплексов программ охватывают столь разнообразные сферы деятельности человека, что невозможно выделить и унифицировать небольшое число атрибутов для оценки и сравнения этой характеристики в различных комплексах программ.

Рекомендации по оценке качества ИС

- ◎ *Оценка корректности программных средств* — формальное определение степени соответствия комплекса реализованных программ исходным требованиям контракта, технического задания и спецификаций на программное средство и его компоненты.

Путем верификации должно быть определено соответствие исходным требованиям всей совокупности компонентов комплекса программ, вплоть до модулей и текстов программ и описаний данных.

Рекомендации по оценке качества ИС

- ◎ *Оценка способности к взаимодействию* – определение качества совместной работы компонентов программных средств и баз данных с другими прикладными системами и компонентами на различных вычислительных платформах, а также взаимодействия с пользователями в стиле, удобном для перехода от одной вычислительной системы к другой с подобными функциями.

Рекомендации по оценке качества ИС

- ◎ *Оценка защищенности программных средств* – определение полноты использования доступных методов и средств защиты программного средства от потенциальных угроз и достигнутой при этом безопасности функционирования информационной системы.

Наиболее широко и детально методологические и системные задачи оценки комплексной защиты информационных систем изложены в стандарте ISO 15408:1999-1--3 "Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий".

Рекомендации по оценке качества ИС

- ◎ *Оценка надежности* - измерение количественных метрик характеристик в использовании: завершенности, устойчивости к дефектам, восстанавливаемости и доступности/готовности.

Рекомендации по оценке качества ИС

- ◎ *Потребность в ресурсах памяти и производительности* компьютера в процессе решения задач значительно изменяется в зависимости от состава и объема исходных данных.

Для корректного определения предельной пропускной способности ИС нужно измерить экстремальные и средние значения времени исполнения функциональных групп программ и маршруты, на которых они достигаются.

Если предварительно в процессе проектирования производительность компьютера не оценивалась, то, скорее всего, понадобится большая доработка или даже замена компьютера на более быстродействующий.

Рекомендации по оценке качества ИС

- ◎ *Оценка практичности* программных средств проводится экспертами и включает определение понятности, простоты использования, изучаемости и привлекательности программного средства.

В основном это качественная (и субъективная) оценка в баллах, однако некоторые атрибуты можно оценить количественно: трудоемкость и длительность выполнения операций при использовании программного средства, объем документации, необходимой для их изучения.

Рекомендации по оценке качества ИС

- ◎ *Сопровождаемость* оценивается полнотой и достоверностью документации о состояниях программного средства и его компонентов, всех предполагаемых и выполненных изменениях, позволяющей установить текущее состояние версий программ в любой момент времени и историю их развития.

Рекомендации по оценке качества ИС

- ◎ *Оценка мобильности* - качественное определение экспертами адаптируемости, простоты установки, совместимости и замещаемости программ, выражаемое в баллах.

Количественно эту характеристику программного средства оценивают экономическими показателями: стоимость, трудоемкость и длительность реализации процедур переноса на иные платформы определенной совокупности программ и данных.

В настоящее время не существует стандартов, полностью удовлетворяющих оценке качества ИС.

Международные стандарты серии ISO 9000 являются методической основой для управления качеством, при этом:

- они определяют и регламентируют инвариантные вопросы создания, развития, применения и сертификации систем качества в промышленности;
- в них устанавливается форма требований к системе качества в целях демонстрации поставщиком своих возможностей и оценки этих возможностей внешними сторонами.

В западноевропейских странах имеется ряд стандартов, определяющих основы сертификации программных систем

Стандарты серии ISO 9000 дополняют стандарты серии ISO 14000,

отражающие экологические требования к производству промышленной продукции, которые:

- задают процедуры управления и контроль документации;
- аудит;
- подразумевают соответствующее обучение.

Кроме требований заказчиков и покупателей, в них воплощаются внутренние требования организации.

Оценка эффективности ИС

С расширением применения ИТ эффективность ИС предприятия целесообразно рассматривать с точки зрения эффективности внедрения ИС и технологий (ИТ-проекта внедрения ИС).

Экономическая эффективность ИТ-проекта характеризуется системой экономических показателей, отражающих соотношение связанных с проектом затрат и результатов и позволяющих судить об экономической привлекательности проекта для его участников, об экономических преимуществах одних проектов перед другими.

Инструменты оценки эффективности внедрения ИС



Реинжиниринг и его место в ЖЦ ИС.

Методы и технологии
реинжиниринга ИС

Сегодня можно говорить, что время, когда разработчики ИС приходили в организацию и начинали проекты информатизации «с нуля», прошла.

Наступило время проектов по систематической трансформации существующих ИС или

эра реинжиниринга ИС.

Основные понятия

Инжиниринг бизнеса — это набор приемов и методов, которые компания использует для проектирования бизнеса в соответствии со своими целями.

Реинжиниринг — это *фундаментальное* переосмысление и *радикальное* перепроектирование деловых *процессов* для достижения *резких, скачкообразных улучшений* главных современных показателей деятельности компании, таких, как стоимость, качество, сервис и темпы
(термин «реинжиниринг» ввел М. Хаммер).

Определения бизнес-процессов:

Бизнес-процесс – несколько связанных работ или процедур, в совокупности реализующих конкретную цель текущей деятельности в рамках существующей организационной структуры.

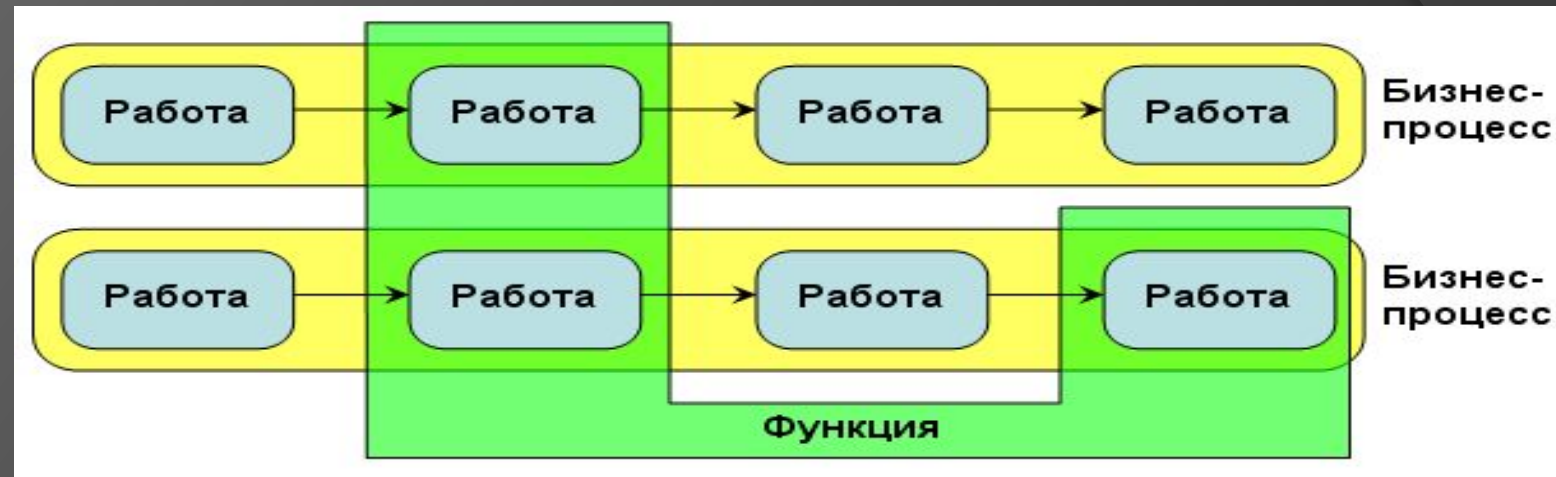
Бизнес-процесс – совокупность различных видов деятельности, в рамках которой «на входе» используется один или более видов ресурсов, и в результате этой деятельности «на выходе» создаётся продукт, представляющий ценность для потребителя

Бизнес-процесс – совокупность взаимосвязанных или взаимодействующих видов деятельности, которые преобразуют «ВХОДЫ» в «ВЫХОДЫ» (ISO 9000:2000)



Бизнес-процессы и Функции

Общие характеристики



- Бизнес-процессы и функции состоят из работ;
- Различие между процессами и функциями заключается в различии способов объединения работ и используемых элементов для их описания.
- Функция является частью бизнес-процесса и при этом может входить как в различные этапы процесса, так и в различные бизнес-процессы.

Реинжиниринг обладает следующими свойствами:

- ◎ **отказ от устаревших правил и подходов** и начало делового процесса как бы «с чистого листа». Это позволяет преодолеть негативное воздействие сложившихся хозяйственных догм;
- ◎ **пренебрежение действующими системами, структурами и процедурами** предприятия и радикальное изменение способов хозяйственной деятельности;
- ◎ если невозможно переделать свою деловую среду, то можно **переделать свой бизнес**;
- ◎ **приведение к значительным изменениям показателей** деятельности, на порядок отличающихся от предыдущих.

Направления перепроектирования бизнес-процессов:

- несколько рабочих процедур, выполнявшихся ранее различными сотрудниками, объединяются в одну — горизонтальное сжатие процесса;
- исполнитель принимает самостоятельные решения — вертикальное сжатие процессов;
- распараллеливание процесса там, где это возможно;
- реализация процессов в различных вариантах исполнения, там где это необходимо;
- выполнение процесса в том месте, где это целесообразно;
- уменьшение количества проверок и управляющих воздействий, которые непосредственно не производят материальных ценностей;
- минимизация количества согласований, которые не производят непосредственных ценностей для заказчика;
- единая точка контакта за счет назначения специального менеджера;
- преобладание смешанного централизованно/децентрализованного подхода.

Направления использования ИТ в реинжиниринге бизнес-процессов:

- ◎ применение методов ИТ для анализа и конструирования бизнес-процессов;
- ◎ создание новых бизнес-процессов, позволяющих коренным образом изменить базовые правила работы организаций;
- ◎ разработка новой ИС.

Факторы, влияющие на успешность реализации проекта по реинжинирингу

- 1. Поддержка проекта высшим руководством от начала до конца;**
- 2. Учет стратегических целей;**
- 3. Наличие квалифицированных экспертов;**
- 4. Понимание своей роли в процессе реинжиниринга персоналом предприятия.**

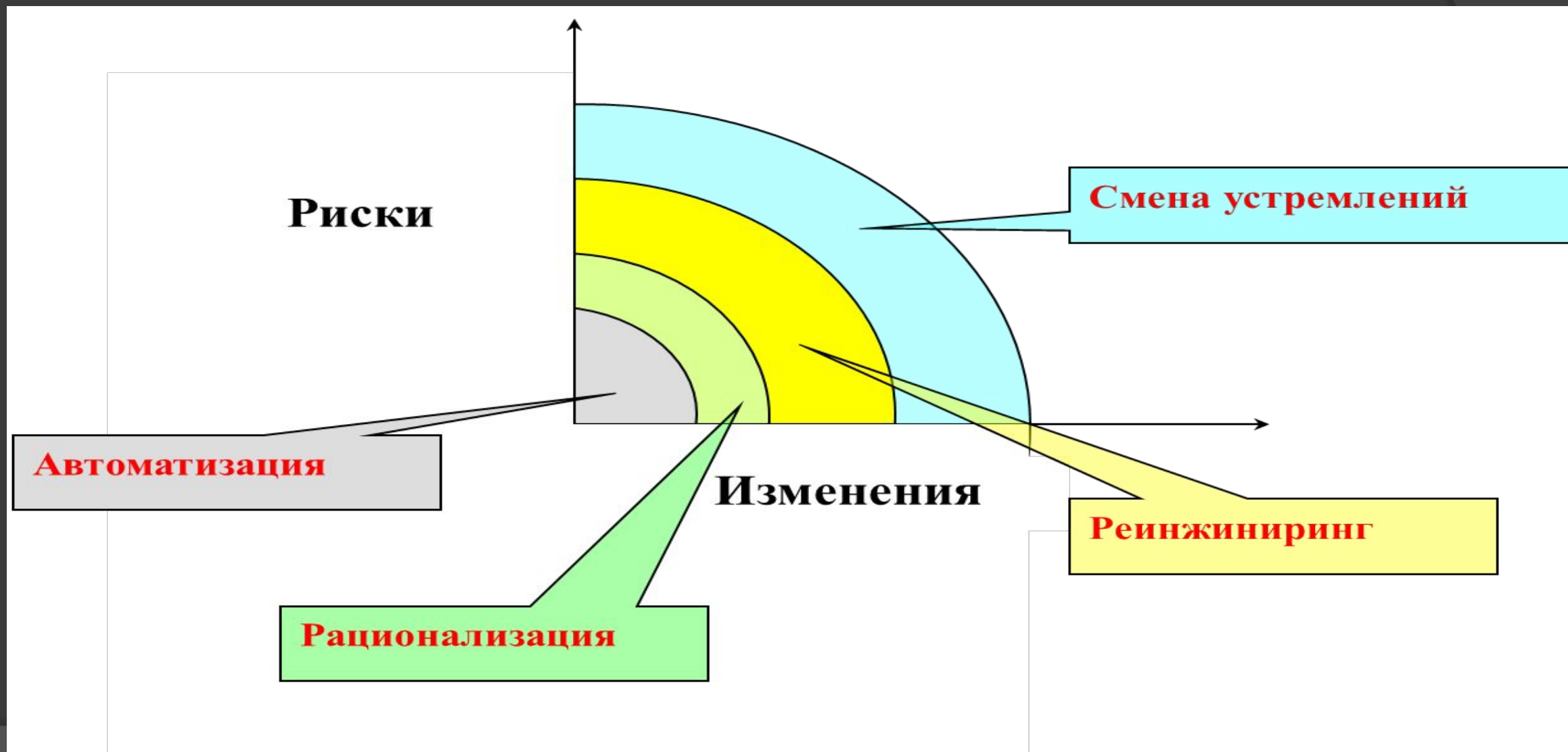
Факторы, препятствующие успешной реализации проекта по реинжинирингу

- 1. Отсутствие поддержки со стороны управляющего персонала;**
- 2. Смена стиля руководства в ходе выполнения проекта;**
- 3. Недостаточная компетенция специалистов, выполняющих проект.**

Сразу следует признать, что в настоящий момент понятие «реинжиниринг ИС» не является повсеместно устоявшимся.

в качестве базовых понятий, наряду с «реинжиниринг ИС» употребляются «эволюция ИС», «миграция ИС», «модернизация ИС», «реструктуризация ИС».

Структурные изменения посредством ИТ



Реинжиниринг представляет собой систематическую трансформацию существующей системы с целью улучшения ее характеристик качества, поддерживаемой ею функциональности, понижения стоимости ее сопровождения, вероятности возникновения значимых для заказчика рисков, уменьшения сроков работ по сопровождению системы.

Подходы, методы и технологии миграции, модернизации, эволюции ИС следует считать частью методологического и инструментально - технологического обеспечения процесса реинжиниринга ИС!

В контексте деятельности по реинжинирингу

ВВОДЯТСЯ следующие понятия

- ◎ прямой инжиниринг (Forward engineering);
- ◎ редокументирование (Redocumentation);
- ◎ рефакторинг (Refactoring);
- ◎ реструктуризация (Restructuring);
- ◎ переориентация (Retargeting);
- ◎ обратный инжиниринг (обратное проектирование) (Reverse engineering);
- ◎ сопровождение программных продуктов (Software maintenance);
- ◎ трансляция исходного кода (Source Code Translation);
- ◎ и т.д.

Перечисленные понятия раскрывают понятие «реинжиниринг ИС», а соотносимая с ними деятельность рассматривается либо как одна из форм реинжиниринга ИС, либо как подпроцесс реинжиниринга.

Утверждается, что реинжиниринг ИС занимает промежуточное местоположение по отношению к разработке и сопровождению ИС.

При этом **сопровождение ИС** рассматривается как деятельность, предусматривающая выполнение изменений, направленных на коррекцию, усовершенствование и адаптацию ИС,

разработка ИС как деятельность, включающая реализацию новых возможностей, добавление новой функциональности, осуществление таких существенных улучшений, как переход на использование новых компьютеров, внедрение новых информационных технологий.

В контексте исследований, связанных с эволюцией ИС, выделяются **деятельности по сопровождению, модернизации и замещению ИС.**

◎ **Сопровождение ИС** представляет собой пошаговый итеративный процесс, в рамках которого выполняются малые изменения в системе, не затрагивающие ее структурной организации (архитектуры системы).

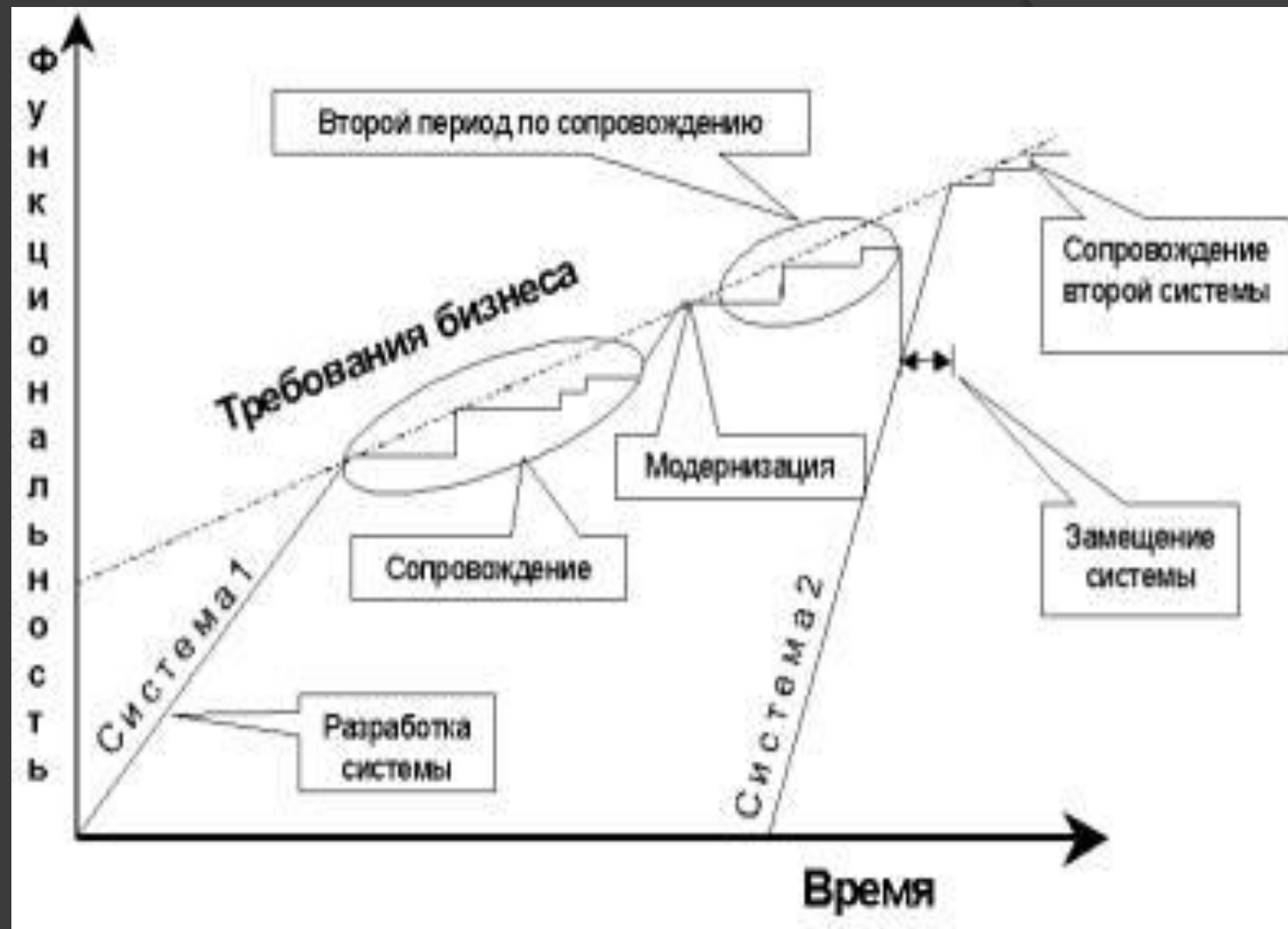
- ◎ В отличие от сопровождения **модернизация ИС** характеризуется как деятельность, которая предусматривает **значительные изменения существующей системы** (в том числе в ее структуре), но не ее утилизацию или **замещение новой системой**.

◎ **Замещение ИС** рассматривается как процесс, который заключается во внедрении новой системы, способной полностью заменить существующую ИС.

Замещение обычно применяется к системам, которые недокументированы, устарели или не расширяемы, расходятся с требованиями бизнеса и для которых модернизация невозможна или экономически не выгодна.

Жизненный цикл ИС

Определяя место видов деятельности реинжиниринга в контексте ЖЦ ИС, рассматривается следующая последовательность их выполнения



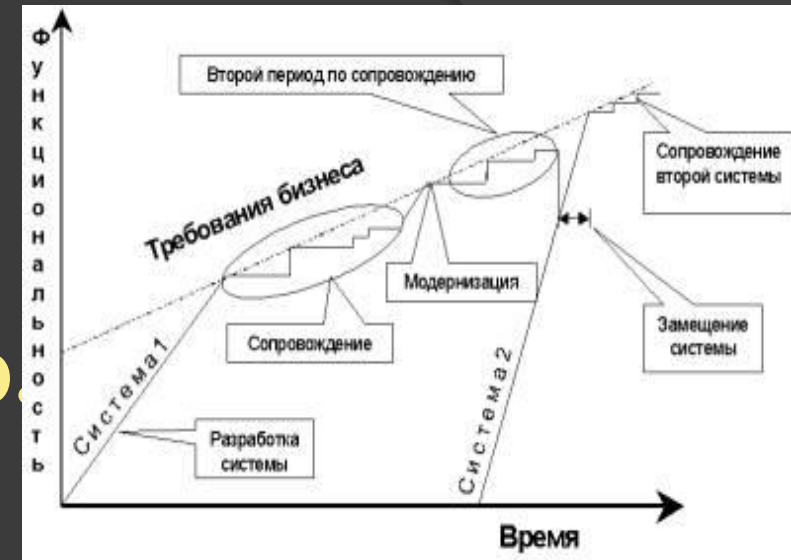
Первоначально, осуществляется **разработка (построение) ИС.**

Далее выполняется **деятельность по ее сопровождению.**

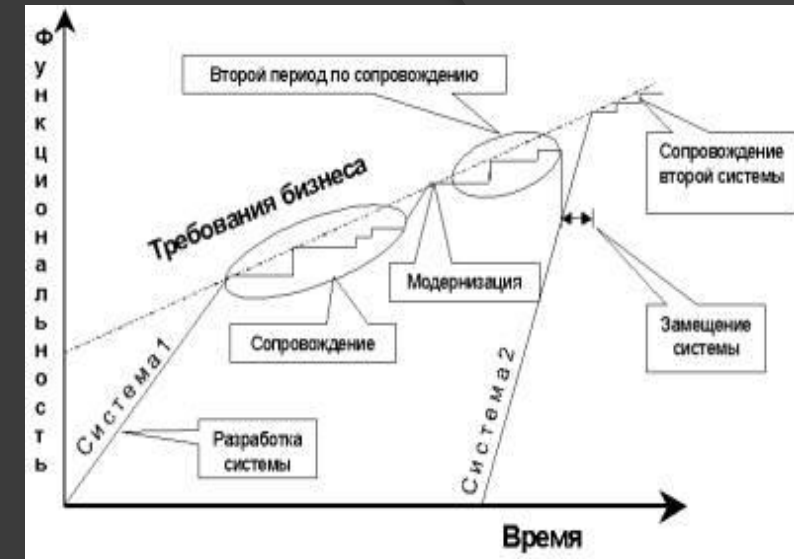
В процессе сопровождения

возникает необходимость в реструктуризации системы и как следствие выполняется **деятельность по ее модернизации.**

После этого снова осуществляется **деятельность по сопровождению системы.**



В тот момент, когда ИС перестает удовлетворять требованиям заказчика, осуществляется **замещение на новую систему и последовательность выполняемых видов деятельности повторяется.**



В реальной жизни может возникать и другая последовательность, когда, сразу, не прибегая к модернизации, осуществляется замещение одной ИС другой.

Однако общий подход к последовательности выполнения сопровождения, модернизации и замещения останется неизменным.

Основные виды деятельности (фазы) проведения реинжиниринга:

- оценка показателей проекта по реинжинирингу, в том числе характеристик унаследованной информационной системы (фаза оценки);
- анализ решений по реинжинирингу, в том числе принятие решения о необходимости проведения работ по реинжинирингу или сопровождению/разработке ИС;
- осуществление реинжиниринга (выполнение работ по реинжинирингу);
- внедрение системы, трансформированной в результате проведения реинжиниринга.

Другой подход к определению деятельности по реинжинирингу базируется на так называемой модели «подкова». В основу данной модели положены следующие процессы (виды деятельности), соотносимые с реинжинирингом ИС:

- анализ существующей системы, основанный на одном или более ее логических описаний;
- трансформация этих логических описаний в новое, улучшенное логическое описание системы.
- разработка новой системы, основанной на новых логических описаниях системы.

Основные процессы модели в виде «ПОДКОВЫ»



Следует признать, что **модель «подкова»** находит **широкое применение в рамках деятельности, связанной с реинжинирингом ИС.**

На ее основе определяются требования и основной каркас для интеграции инструментальных средств реинжиниринга на архитектурном уровне и уровне программного кода.

С учетом соотносимых с ней процессов и уровней представления, осуществляется расширение модели CORUM (Common Object-based Reengineering Unified Model).

Модель «Подкова» разрабатывалась:

- для представления на основе программного кода (Code-based Management System) требуемой для систем управления информации о программных средствах;
- для обеспечения интероперабельности между системами данного класса (в том числе между средствами реинжиниринга программ на уровне программного кода).

Этапы процесса реинжиниринга:

- ◎ формирование команды реинжиниринга;
- ◎ анализ требований для выявления конкретных целей реинжиниринга унаследованной системы;
- ◎ восстановление модели, в том числе документирование и понимание структуры унаследованной системы;
- ◎ выявление проблем, связанных с унаследованной системой;
- ◎ анализ проблем, включающий выбор архитектуры, позволяющий устранить обнаруженные в унаследованной системе дефекты;
- ◎ реорганизация, включающая выбор и применение оптимального подхода трансформации унаследованной системы;
- ◎ распространение изменений.

Общепринято выделять следующих участников процесса реинжиниринга:

- ⦿ лидер проекта;
- ⦿ руководящий комитет наблюдателей;
- ⦿ специалист, отвечающий за развитие методик и инструментариев поддержки реинжиниринга;
- ⦿ команда по реинжинирингу ;
- ⦿ эксперт по методу отвечает за используемую методологию;
- ⦿ группа обеспечения качества;
- ⦿ группа документирования;
- ⦿ координатор (или группа координации);
- ⦿ лидеры процессов и владельцы процессов;
- ⦿ штат менеджера проекта.

Методы и технологии реинжиниринга ИС

В настоящий момент существует значительное количество литературы, посвященной проблемам, методам и технологиям реинжиниринга ИС.

Эти работы охватывают данную проблематику с различных точек зрения, рассматривая и исследуя в различной степени как проблемы концептуального уровня, так и конкретные методы, и инструментальные средства, предназначенные для реинжиниринга ИС.

Проблемы применения методов и технологий реинжиниринга ИС

Несмотря на наличие множества различных решений, их исследование и комплексное применение на практике бывает затруднено. Причинами возникающих трудностей следует считать:

- ⦿ понятие «реинжиниринг ИС» до сих пор трактуется по-разному, существует множество близких понятий, наличие которых приводит к появлению внешне отличающихся, но по сути схожих подходов, методов и технологий;
- ⦿ предлагаемые решения не позиционируются в контексте других существующих решений;

Проблемы применения методов и технологий реинжиниринга ИС

- ⦿ решения не интегрированы на уровне методологий и технологий, большое количество методов и инструментальных средств направлено на решение отдельных локальных задач, связанных с реинжинирингом ИС;
- ⦿ наблюдается разрыв между решениями концептуального характера и решениями, направленными на решение конкретных прикладных задач.

Классифицируя существующие подходы, методы и технологии, можно выделить следующие **уровни рассмотрения и исследования аспектов, соотносимых с деятельностью по реинжинирингу ИС**

УРОВЕНЬ 6 Технология реинжиниринга ИС	УРОВЕНЬ 5 Методология реинжиниринга ИС	УРОВЕНЬ 1 Понятие «реинжиниринг ИС», место реинжиниринга в ЖЦ ИС
		УРОВЕНЬ 2 Процесс реинжиниринга ИС, связи между шагами процесса
		УРОВЕНЬ 3 Методы решения локальных задач, возникающих при реинжиниринге ИС
		КЛАСС 1 Методы, не зависящие от конкретной программной технологии
		КЛАСС 2 Методы, ориентированные на использование в рамках конкретной программ. технологии
		УРОВЕНЬ 4 Инструментальные программные средства

◎ *Первый уровень* включает исследования, направленные на достижение концептуального понимания деятельности по реинжинирингу ИС. Именно на этом уровне исследуются вопросы адекватного определения понятия «реинжиниринг ИС», определения места реинжиниринга в жизненном цикле ИС, в том числе выявление связей процесса реинжиниринга ИС в целом с другими процессами ЖЦ ИС.

Следует считать, что большая часть рассмотренных ранее аспектов, относятся к этому уровню.

**Уровни
рассмотрения и
исследования
аспектов
реинжиниринга ИС**

УРОВЕНЬ 6 Технология реинжиниринга ИС	УРОВЕНЬ 5 Методология реинжиниринга ИС	УРОВЕНЬ 1 Понятие «реинжиниринг ИС», место реинжиниринга в ЖЦ ИС
		УРОВЕНЬ 2 Процесс реинжиниринга ИС, связи между шагами процесса
		УРОВЕНЬ 3 Методы решения локальных задач, возникающих при реинжиниринге ИС
		КЛАСС 1 Методы, не зависящие от конкретной программной технологии
		КЛАСС 2 Методы, ориентированные на использование в рамках конкретной программ. технологии
	УРОВЕНЬ 4 Инструментальные программные средства	

◎ *Второй уровень* содержит исследования, основная цель которых заключается в выявлении основных шагов (действий), реализуемых в процессе реинжиниринга и в определении связей между основными шагами процесса.

Здесь в сферу рассмотрения попадают потоки управления и потоки данных между основными шагами процесса, основные роли, соотносимые с исполнителями процесса, а так же правила распределения ролей среди команды исполнителей.

**Уровни
рассмотрения и
исследования
аспектов
реинжиниринга ИС**

УРОВЕНЬ 6 Технология реинжиниринга ИС	УРОВЕНЬ 5 Методология реинжиниринга ИС	УРОВЕНЬ 1 Понятие «реинжиниринг ИС», место реинжиниринга в ЖЦ ИС
		УРОВЕНЬ 2 Процесс реинжиниринга ИС, связи между шагами процесса
		УРОВЕНЬ 3 Методы решения локальных задач, возникающих при реинжиниринге ИС
		КЛАСС 1 Методы, не зависящие от конкретной программной технологии
		КЛАСС 2 Методы, ориентированные на использование в рамках конкретной программ. технологии
	УРОВЕНЬ 4 Инструментальные программные средства	

◎ *На третьем уровне* рассматриваются (исследуются и разрабатываются) методы, каждый из которых направлен на решение некоторой локальной задачи, возникающей в процессе реинжиниринга ИС, например, выполнения определенного шага процесса.

По сути, эти методы воплощают собой некоторые вполне конкретные решения, с которыми соотносится определенная область применения.

Примером процесса, направленного на решение локальной задачи, является итеративный процесс, определяющий следующую **последовательность шагов, которые должны быть выполнены при планировании проектов реинжиниринга ИС:**

- ⦿ Определение целей, направлений деятельности организации и информационной системы.
- ⦿ Формирование объединенной команды, которая будет осуществлять реинжиниринг унаследованной системы.

Последовательность шагов, которые должны быть выполнены при планировании проектов реинжиниринга ИС:

- ⦿ Определение среды разработки и сопровождения, базирующейся на применении модели оценки уровня зрелости ИТ-инфраструктуры.
- ⦿ Выбор стандартного множества метрик для оценки программных средств.
- ⦿ Анализ унаследованной системы.
- ⦿ Определение процесса реализации деятельности по реинжинирингу.
- ⦿ Разработка/Обновление стандартных средств тестирования и валидации.
- ⦿ Анализ средств реинжиниринга.
- ⦿ Обучение.

Условно, все методы реинжиниринга ИС можно разделить на два класса.

◎ Методы, относящиеся к первому классу, определены на концептуальном уровне и в целом не зависят от какой-то одной программной технологии.

Среди всего множества рассматриваемых методов к первому классу относятся метод репликации баз данных и основанный на объектно-ориентированном подходе метод построения оболочек для компонентов унаследованной системы (object – oriented wrapping), методы «белого» и «черного» ящика модернизации системы.

Методы реинжиниринга ИС

- К другому классу относятся: методы оценки вариантов реинжиниринга ИС, метод планирования миграции программных средств, методы извлечения знаний о существующей системе, методы трансформации (реконструкции) архитектуры ИС, методы автоматизации реинжиниринга программ и т.д.

В целом следует считать, что область применения таких методов, как правило, характеризуется некоторым классом программных технологий. А для применения в реальных проектах каждый из них адаптируется с учетом используемых в проекте технологий и инструментальных средств.

При проектировании используются различные методологии и соответствующие нотации:

- ◎ SADT: IDEF0, IDEF3, DFD
- ◎ ORACLE Diagram
- ◎ BAAN Diagram
- ◎ ARIS Value Added Chain Diagram
- ◎ ARIS eEPC Diagram
- ◎ БИТЕК Диаграмма бизнес-процесса
- ◎ BPMN
- ◎ RUP
- ◎ UML(Unified Modeling Language):
Унифицированный язык моделирования

Программные продукты, реализующие методологию IDEF0:

- ◎ Business Studio
- ◎ CA AllFusion Process Modeler 7 (ранее VPwin)
- ◎ Corel iGrafx (IDEF0)
- ◎ Design/IDEF 3.5
- ◎ Casewise Corporate Modeler 10
- ◎ IDEF0\Doctor
- ◎ IDEF0 on WEB
- ◎ IDEF0.EM Tool
- ◎ MS Visio

Опыт реализации реинжиниринга бизнес-процессов

Опыт IBM Credit Corporation, филиал IBM: занимается кредитованием клиентов, которым IBM продает компьютеры, программы и предоставляет услуги.

Проблема: при существующем технологическом цикле **решение вопроса о кредитовании клиента занимало в среднем 7 дней, а в сложных случаях — до двух недель.** Чрезмерная длительность принятия решений приводила к **потере клиента**, так как он за это время мог найти другой источник финансирования.

Длительность принятия решения по запросу клиента: обработка запроса осуществлялась в 5 шагов, выполняемых последовательно в пяти различных подразделениях компании. При этом передача запроса из одного подразделения в другое осуществлялось на бумажном носителе.

В новом процессе всю обработку выполняет один специалист, снабженный информационной экспертной системой, обеспечивающей принятие решения и доступ ко всем необходимым данным и инструментариям. Теперь в большинстве случаев (более 90% запросов) один специалист обеспечивает решение задачи. В трудных случаях специалист обращается к экспертам.

Результат: время обработки запроса сокращено с 7 дней до 4-х часов, количество обрабатываемых запросов возросло в 100 раз при уменьшении количества сотрудников!!!

Опыт реализации реинжиниринга бизнес-процессов

Опыт Ford Motor: решено сократить расходы в отделении оплаты счетов, где работало более 500 человек.

Отделение счетов само по себе не могло быть подвергнуто реинжинирингу, так как это подразделение, а не процесс.

Процесс Поставки: департамент заказов посылает продавцу товаров заказ на их приобретение; копия заказа отправляется в отделение оплаты счетов; когда товары прибыли в компанию, сотрудник из отдела получения товаров составляет документ получения и отправляет его в департамент оплаты счетов; продавец посылает в отделение оплаты счетов накладную на товары.

К этому времени в отделении оплаты счетов находятся 3 документа на эти товары: заказ на приобретение, документ получения и накладная. Если все три документа соответствуют друг другу (в большинстве случаев именно эта ситуация и имеет место), то оплачивается счет. При несоответствии документов необходимо найти источник ошибки. Основное время в своей работе сотрудник тратит на обработку именно таких ситуаций.

Способ предотвратить расхождения: ввели «обработку без счетов» — когда отдел закупок дает заказ, информация о нем вводится в базу данных, копии заказа никому не посылаются; когда товар прибывает на разгрузку, приемщик проверяет базу данных, чтобы выяснить, соответствует ли этот товар какому-либо неисполненному заказу. Если да, производится приемка, сведения о которой также заносятся в базу данных, если нет, заказ просто возвращается поставщику.

Результат: количество сотрудников уменьшилось на 75 %, а не на 20 %, как прогнозировалось ранее. Поскольку теперь не существует разночтений между финансовым и материальным учетом, входной контроль значительно проще, а финансовая информация — точнее.

Основная задача любого успешного проекта

обеспечить:

- ⦿ требуемую функциональность системы и степень адаптации к изменяющимся условиям ее функционирования;
- ⦿ требуемую пропускную способность системы;
- ⦿ требуемое время реакции системы на запрос;
- ⦿ безотказную работу системы в требуемом режиме, то есть: готовность и доступность системы для обработки запросов пользователей;
- ⦿ простоту эксплуатации и поддержки системы;
- ⦿ требуемую безопасность
- ⦿ высокую эффективность

**СПАСИБО ЗА
ВНИМАНИЕ!**